Kolmogorov 記述量に基づく類似度を用いた方言の自動分類

Automatic classification of Japanese dialects using similarity metric is based on Kolmogorov Complexity

谷　研究室　　堀中　幸司
Koji Horinaka

概要

Kolmogorov 記述量に基づいたデータ間の類似度に関する距離が定義され、DNA の類似度や言語の類似度、音楽の類似度判定に有用だという実験結果が得られている。本研究では Kolmogorov 記述量が方言の類似度分析に対して有用かどうか実験を行なった大江氏の追加研究を行なう。

1　はじめに

現在の方言研究というものは単語単位、アクセント、イントネーションによる研究がなされている。しかし文章単位での研究というものは余り進められていない。そこで近年 Ming Li らが Kolmogorov 記述量に基づくデータ間の類似度に関する距離を表す similarity metric を発案（現在 DNA の類似度や言語の類似度、音楽の類似度判定に有用だということが分かっている。）により、昨年大江氏が「Kolmogorov 記述量に基づく類似度を用いた方言の自動分類」の研究をした。本研究はその追加研究を行なう。

本研究では、音声データをテキスト化をするのだが、文字コードの違い、前処理を加える事による違い、クラスタリング（NJ、UPGMA、Quartet Method）による違いなどで検証を進めていく。

第２節では Kolmogorov 記述量について、第３節では Similarity metric について、第４節では、NJ 法（近隣結合法）の階層型クラスタリングアルゴリズムの解説（UPGMA,Quartet Method のアルゴリズム解説は同年研究報告書の高野氏、関根氏の報告書を見ること）、第５節では木の評価に関しての定義を、第６節では実験概要、実験データ、入力方法、文字コードによる違いについて、第７節では実験結果、考察、第８節では今後の課題を述べる。

2　Kolmogorov 記述量の定義

Kolmogorov 記述量の基本的な定義を述べておく。

2.1　Kolmogorov 記述量

あるデータ x が存在し、データ x の Kolmogorov 記述量とはあるプログラム言語で x を生成する最小のプログラムのサイズである。どんなプログラム言語を選んでも、それが妥当であれば情報量は定数分の差しかないことが証明されている。S をプログラム言語、$|p|$ をプログラムサイズとすると、データ x の Kolmogorov 記述量 $K_s(x)$ は、以下のように定義される。

$$K_s(x) = min|p| : S(p) = x$$

すなわち $K_s(x)$ は、”プログラム言語 S において x を生成する最小のプログラムの長さ ”と考えていいだろう。

次に対象 y が与えられているときの対象 x についての記述量（相対記述量）について考える。文字列 x を、計算可能関数（インタプリタ関数）f、それに文字列 p と y により $f(p, x) = x$ と記述することを考える。インタプリタ関数 f のもとで、補助関数 y に対する x の記述量 K_f を次のように定義する。

$$K_f(x|y) = min\{|p| : P \in \{0,1\} * f(p, y) = x\}$$

また、x と y を区別できる形で出力させる最短のプログラムの長さ $K(xy)$ とあらわす。$O(logK(xy))$ の誤差範囲では、

$$K(xy) = K(x) + K(y|x)$$

となることが証明されている。

2.2　K のアルゴリズム的性質

$K(x)$ を正の整数から正の整数への関数と考える。

定理

(a) 関数 $K(x)$ は帰納的ではない。しかも、どのような機能的部分関数 $\phi(x)$ を考えても、もしその値が無限個の点で定義されているのならば、その定義域上のどこかの点で $\phi(x) \neq K(x)$ となる。

(b) 引数 t に対しては単調減少で（全域的な）機能的関

数 $H(t,x)$ が存在し,$\lim_{t\to\infty} H(t,x) = K(x)$。すなわち、$K(x)$ の良い近似（ただし一様近似ではない）は計算不可能。

2.3 情報量

相対記述量 $K(x|y)$ が絶対記述量 $K(x)$ よりかなり小さい場合、「y が x についての情報をたぶんに含んでいる」と考えることが出来る。したがって、定数分の差異を無視すれば、”y に含まれる x の情報量 ”は

$$I(x:y) = K(y) - K(y|x)$$

とみなすことが出来る。また、$K(x|x)$ となる x を選べば、

$$I(x:x) = K(x)$$

となる。ここで情報の対象性ということを考える。$OlogK(xy)$ の誤差範囲では、

$$K(xy) = K(x) + K(y|x)$$

である。したがって

$$I(x:y) = I(y:x)$$

が成立する。

3 similarity metric

定義

ある集合 X 上の距離とは、実数値関数 $d : X \times X \to R$ で任意の $x, y, z \in X$ に対して次のような性質を満たす。

$$d(x,y) \geq 0$$
$$d(x,y) = 0 \Leftrightarrow x = y$$
$$d(x,y) = d(y,x)$$
$$d(x,y) \leq d(x,z) + d(z,y) : \text{三角不等式}$$

これを基に、情報の距離を考える。

Ming Li らの研究では、正規化情報距離が提案されている。任意の文字列 x,y について、以下のように定めている。

$$d(x,y) = \frac{max\{K(x|y), K(y|x)\}}{max\{K(x), K(y)\}}$$

また $K(y) \geq K(x)$ としたとき、

$$d(x,y) = \frac{K(y|x)}{K(y)}$$

これに対して先に述べた情報量の公式を用いると、

$$d(x,y) = \frac{K(y) - I(x:y)}{K(y)}$$

となる。

また $O(logK(xy))$ の範囲では $K(xy) = K(x) + K(y|x)$ が成り立つので、

$$d(x,y) = \frac{K(xy) - K(x)}{K(y)}$$

と表すことが出来る。

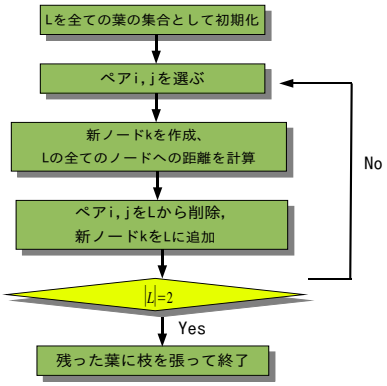
Kolmogorov 記述量は計算可能ではない。擬計算可能ではあるが、実用的な効率の良いアルゴリズムはしられていない。そこで、現実の問題に適用する場合は、圧縮プログラムを利用し、圧縮後のサイズを Kolmogorov 記述量の近似値として用いる。圧縮率が高い圧縮プログラムほど、Kolmogorov 記述量により良く近似できるが、圧縮率が高いプログラムが類似度の距離計算においてよい結果を与えるとは限らない。現在はその圧縮プログラムは bzip2 が有用だとされている。bzip2(x) は対象 x を bzip2 で圧縮したときのファイルサイズをする。

$$d(x,y) \approx \frac{bzip2(xy) - bzip2(x)}{bzip2(y)}$$

この距離関数を用いて分類化していく。

4 階層型クラスタリング

4.1 NJ 法



Step1

L を全ての葉の集合として初期化。

Step2

次の D_{ij} が最小となるペア i, j を選ぶ。

$$D_{ij} = d_{ij} - (r_i + r_j)$$
$$r_i = \frac{1}{|L| - 2} \sum_{k \in L} d_{ik}$$

Step3

新しい接点 k を作り、L のすべての葉 m との距離を求める。

$$d_{km} = \frac{1}{2}(d_{im} + d_{jm} - d_{ij})$$

Step4

k と i, j との距離は、

$$d_{ik} = \frac{1}{2}(d_{ij} + r_i - r_j)$$

$$d_{jk} = d_{ij} - d_{ik}$$

Step5

i, j を L から除き、k を追加、 2へ戻る

Step6

L が i, j の 2 個だけになったら、枝を張って終了

5 木の評価

クラスタリングをして出来た木が similarity metric で求めた類似度の距離に沿って出来ているかを検証するための評価値を定義する。
similarity metric で求めた類似度の距離を $sd(u, v)$ とし、最小値を $min(sd)$、最大値を $max(sd)$ とする。またグラフとしてみた場合の頂点同士の距離（2頂点間の辺数）を $td(u, v)$ とし、$td(u, v)$ のうちの最小値を $min(td)$、最大値を $max(td)$ とする。

$td(u, v)$ を類似度の距離に対応させた距離を $ntd(u, v)$ と定義する。そして変換定数を S とする。

$$S = \frac{max(sd) - min(sd)}{max(td) - min(td)}$$

$$ntd(u, v) = min(sd) + S(td(u, v) - min(td))$$

これらから木の評価値を TV_1、TV_2 と 2 種類定義する。L は末端ノードである。

$$TV_1 = \frac{2}{|L(L-1)|} \sum_{u,v \in L} |ntd(u, v) - st(u, v)|$$

$$TV_2 = \sqrt{\sum_{u,v \in L} (ntd(u, v) - st(u, v))^2}$$

TV_1、TV_2 の値が小さいほど、類似度の距離がより反映されたグラフであることがわかる。

6 実験概要

実験データ

方言ももたろう（監修・著：杉藤美代子（音声言語研究所 所長））というソフトに日本各地５６箇所の音声データがある。その音声データをテキスト化（ひらがな・。のみ使用）また「を」→「お」、「へ」→「え」、「は」→「わ」、伸ばす音は前の音の母音をもう一つ加えるという文章の書き方をした。

文字コード

一般に使われている EUC、JIS、SHIFT-JIS、UTF-8 を使用する。なお改行コードは、全て unix に固定させた。EUC、JIS、SHIFT-JIS は 2 バイト、UTF-8 は 3 バイトの文字コードなのでファイルサイズが変わってくるのは当然である。しかし bzip2 に圧縮し simirality metric で類似度の距離を求めると、距離に違いが生じてしまう。そこで圧縮する際に余計な情報が入らないように前処理として、1 バイトのコードに変換をすることにした。

その前処理方法を 2 種類考案した。前処理 1 は 50 音順で 1 から対応させて変換する。前処理 2 は、使用頻度が高い順に 1 から対応させて変換する。

文字コードによって EUC、JIS、SHIFT-JIS、UTF-8、前処理 1 、前処理 2 の 6 種類のデータが存在するが、bzip2 に圧縮するとき最も圧縮できているもの（最も圧縮後のファイルサイズが小さい）ものを今後の研究データとすることにした。なぜなら、kolmogorov 記述量は圧縮した後のファイルサイズが最も小さいものが、より kolmogorov 記述量に近似できるからである。 6 種類では前処理 2 が一番圧縮されていいることが、検証により分かったのでデータは前処理 2 をしたものによりクラスタリングをした。

木の評価

本研究では NJ 法によるクラスタリングを行なった、同時に高野氏、関根氏による UPGMA、Quartet Method でクラスタリングが行なわれた。そこでどのクラスタリングがより良い分類をしているか S (T) や TV で検証を行なうことにした。

7 実験結果

色分け

日本の 56 箇所のデータを 9 地方に分けて色分けをする。

北海道地方 → 赤　　東北地方 → 緑
関東地方 → 青　　中部地方 → 黄色
近畿地方 → ピンク　四国地方 → 紫
中国 → オレンジ　　九州地方 → 水色
沖縄 → 白

結果

NJ 法でクラスタリングを行なった結果

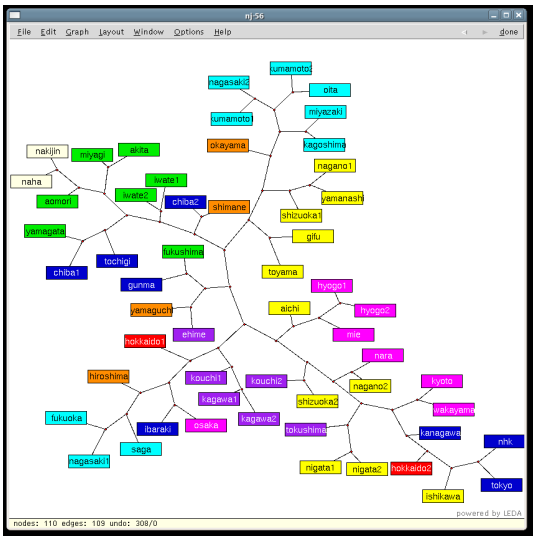


図　nj-remake-preprocess2

木の評価

$S(T)$ を Quartet Method だけではなく UPGMA、NJ 法においても計算をした。$S(T)$ とは Quartet Method でクラスタリングする際に使われる評価値である。その計算結果が下の表である。

ここでは前処理 2 をしたデータでの検証が行なわれた。

（＊ $S(T)$ の詳細については関根氏の報告書を参照）

クラスタリング方法	$S(T)$
Quartet Method	0.850286
NJ 法	0.455565
UPGMA	0.455565

TV_1、TV_2 の評価値は下の表に示す。

クラスタリング方法	TV_1	TV_2
Quartet Method	0.177194	8.268810
NJ 法	0.130395	6.352560
UPGMA	0.162626	7.745910

図を見て分かるように、$S(T)$ は Quartet Method の値が最も良かった、TV_1、TV_2 の値は NJ 法の値が良かった。$S(T)$ が高かった Quartet Method は TV_1、TV_2 の値が低く、類似度の距離が反映されていない。そこで Quartet Method のアルゴリズムに着目した。

Quartet Method はヒューリスティックスであるので、$S(T)$ が同じ 0.85 でも違うグラフが作成されてしまうわけである。つまり TV が低い場合もあれば、高い場合もある。$S(T)$ を更新しつつ、TV の値も評価が高くなるようにクラスタリングを行なえば、$S(T)$、TV 良い値のグラフを作ることが出来るはずであるから、Quartet Method(改) の実験を行なった。

Quartet Method(改) 実験結果 1

$S(T) > 0.85$

TV_2 を更新出来なければ $S(T)$ も更新させない。

実行時間　約 85 時間

$S(T) = 0.780169$

$TV_2 = 5.4471$

TV_2 が小さくなり過ぎて $S(T)$ が更新できないため、グラフが出力出来なかった。

Quartet Method(改) 実験結果 2

$S(T) > 0.85$

$TV_2 < 6.5$

実行時間　約 14 時間

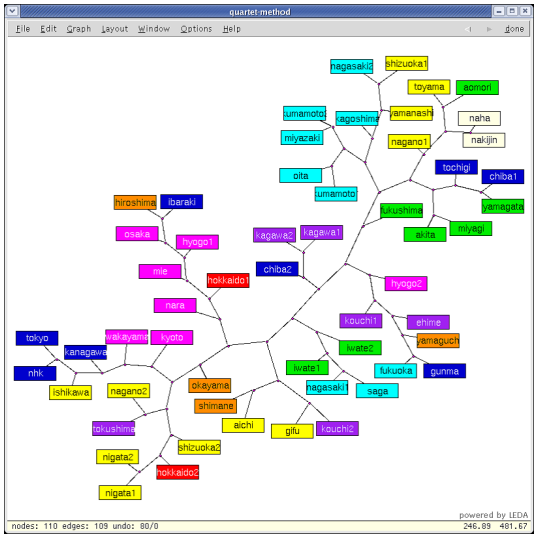


図　$S(T) = 0.850452$　　$TV_2 = 6.46065$

下図は改良を加える前に Quartet Method でクラスタリングをしたグラフである。

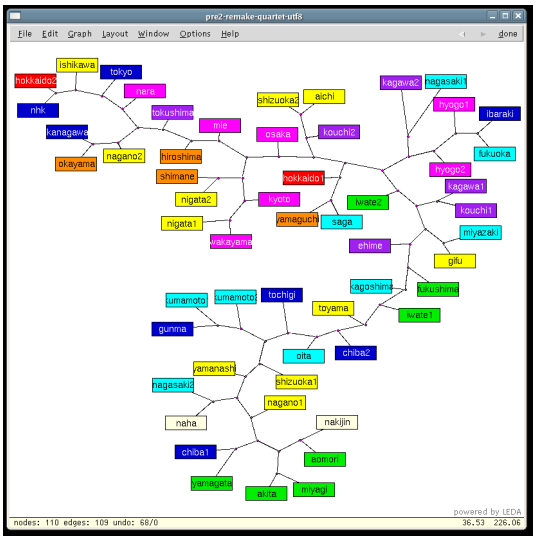


図　$S(T) = 0.850286$　$TV_2 = 8.268810$

２つのグラフを比べてみてもやはり改良型でクラスタリングをした方がより良い分類をしている。

Quartet Method(改) 実験結果（参考）

前処理１のデータの結果 $S(T) > 0.85$

$TV_2 < 5$

実行時間　約 70 時間

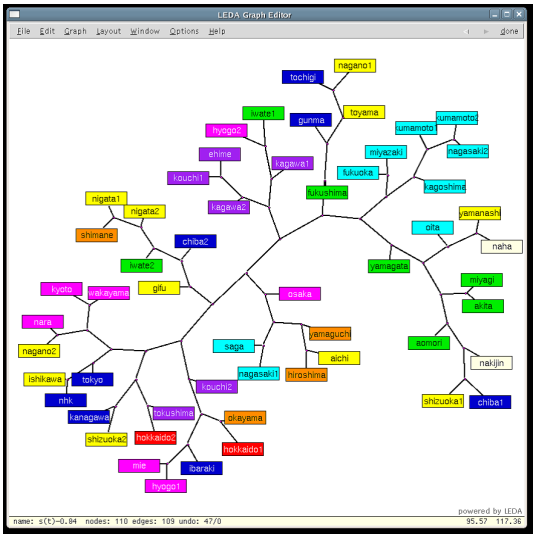


図　$S(T) = 0.84$　$TV_2 = 4.9$

下図は改良を加える前に Quartet Method でクラスタリングをしたグラフである。

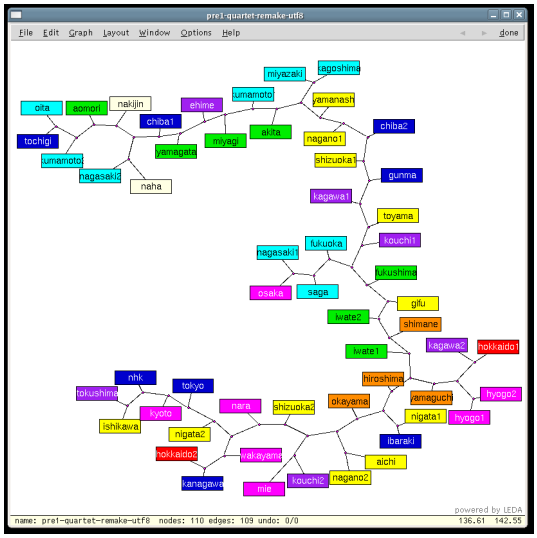


図　$S(T) = 0.85$　$TV_2 = 7.949120$

8　今後の課題

1. アクセントやイントネーションを加える等、対象データの改良する。
2. 音声データでも様々なクラスタリング方法で実験する。
3. 他の階層型クラスタリングを試す。
4. 考案されているクラスタリングを評価する指標を試す。
5. 描画の問題

参考文献

[1] Ming Li and Paul M.B.Vitanyi. 渡辺 治 翻訳 Kolmogorov Complexity and its Applications.　コンピュータ基礎理論ハンドブック (1994)

[2] 谷 聖一　データの複雑さ・データ表現の複雑さ

[3] 大江 碧　Kolmogorov 記述量に基づく類似度を用いた方言の自動分類　2006